J. C. Ferreira, J. L. Afonso, "Mobi_System: A Personal Travel Assistance for Electrical Vehicles in Smart Cities", ISIE 2011 - IEEE International Symposium on Industrial Electronics, Gdansk, Poland, 27-30 June, 2011, pp.1653-1658, ISBN: 978-1-4244-9310-4, doi: 10.1109/ISIE.2011.5984409

Mobi_System: A Personal Travel Assistance for Electrical Vehicles in Smart Cities

João C. Ferreira
GuIAA – ADEETC – ISEL
Lisbon, Portugal
jferreira@deetc.isel.ipl.pt

João L. Afonso
Centre Algoritmi – DEI – University of Minho
Guimarães, Portugal
jla@dei.uminho.pt

***Abstract* - In this work it is proposed the design of a mobile system to assist car drivers in a smart city environment oriented to the upcoming reality of Electric Vehicles (EV). Taking into account the new reality of smart cites, EV introduction, Smart Grids (SG), Electrical Markets (EM), with deregulation of electricity production and use, drivers will need more information for decision and mobility purposes. A mobile application to recommend useful related information will help drivers to deal with this new reality, giving guidance towards traffic, batteries charging process, and city mobility infrastructures (e.g. public transportation information, parking places availability and car & bike sharing systems). Since this is an upcoming reality with possible process changes, development must be based on agile process approaches (Web services).**

Keywords: **Mobile Application; Web Services; Simulation; Electric Vehicle; Energy Market; Smart Grids; Vehicle-to-Grid; Mobile Device**

I. INTRODUCTION

The world population has been progressively concentrating in the cities. Problems associated with urban agglomerations have usually been solved by means of creativity, through infrastructure investments and information technology to support 'smart' solutions. The label 'smart city' should therefore point to clever solutions allowing modern cities to thrive, through quantitative and qualitative improvement in productivity. The concept of the 'smart city' has been quite fashionable in the political arena in recent years. Its main focus seems to be on the role of ICT (Information and Communication Technologies) infrastructure, although much research has also been carried out on the role of human capital/education, social and relational capital, and environmental interest, as important drivers of urban growth. The European Union (EU), in particular, has devoted constant efforts to devising a strategy for achieving urban growth in a 'smart' sense on metropolitan areas. Not only the EU, but also other international institutions, believes in a wired ICT-driven form of development. Connected to this new reality, new paradigms are emerging, like the Electric Vehicle (EV), the Smart Grids (SG), the Vehicle-to-Grid (V2G), and the Electrical Markets (EM). EM is the consequence of the deregulation of electricity production and use, where power suppliers and consumers are free to negotiate the terms of their contracts. Also EVs integration on current electrical distribution network, without violating the electrical system's technical restrictions, requires data on energy consumption analysis and smart charging approaches, where EV batteries charging or discharging processes need to be coordinated among the several users and the energy producers. In this complex scenario, information knowledge related with charging periods, prices, decision of charging or discharging EV batteries, needs assistance and several issues have to be considered and analyzed before taking actions. Although enormous data volumes related with this processes are stored day by day, and hour by hour, it is impossible (through human analysis or with traditional technology) to obtain knowledge from this data, in order to take wise decisions. The aim of the proposed Mobi_System is to improve the transit experience of commuters by giving them access to transportation information via a Web enabled mobile device from any location. Transport information can include data about the most affordable trip plan, minimizing the time of the journey, and Real Time Router which allows the user to optimize their trip, as and when conditions change (eg: traffic accidents). Transportation Information Service can provide information about public transport options, routes, arrival times and schedules.

The research proposal presented in this paper focuses on the creation of a mobile application to help the users to deal with this new reality.

II. MOBI_SYSTEM

Being the next big step in automobile industry, electric vehicles continue to have limited autonomy, which associated with the long charging times, limited number of charging stations, and undeveloped smart grid infrastructures, demands for a hard planning of the daily use of the vehicle. Thus, Information Systems play an important role in information society.

Mobi_System is a personal assistant oriented to smart cites environment and electric vehicles. This upcoming reality uses the integration of public transportation information in real time traffic information, smart grids, open energy market, and smart cities, with increasing mobility sustainability. The proposed system integrates a diversity of functionalities, illustrated on Figure 1:

- Positioning Information: The Mobi_System receives the information system of geographical positioning on the current position of the vehicle and the features that enable the calculation of distances between two points.

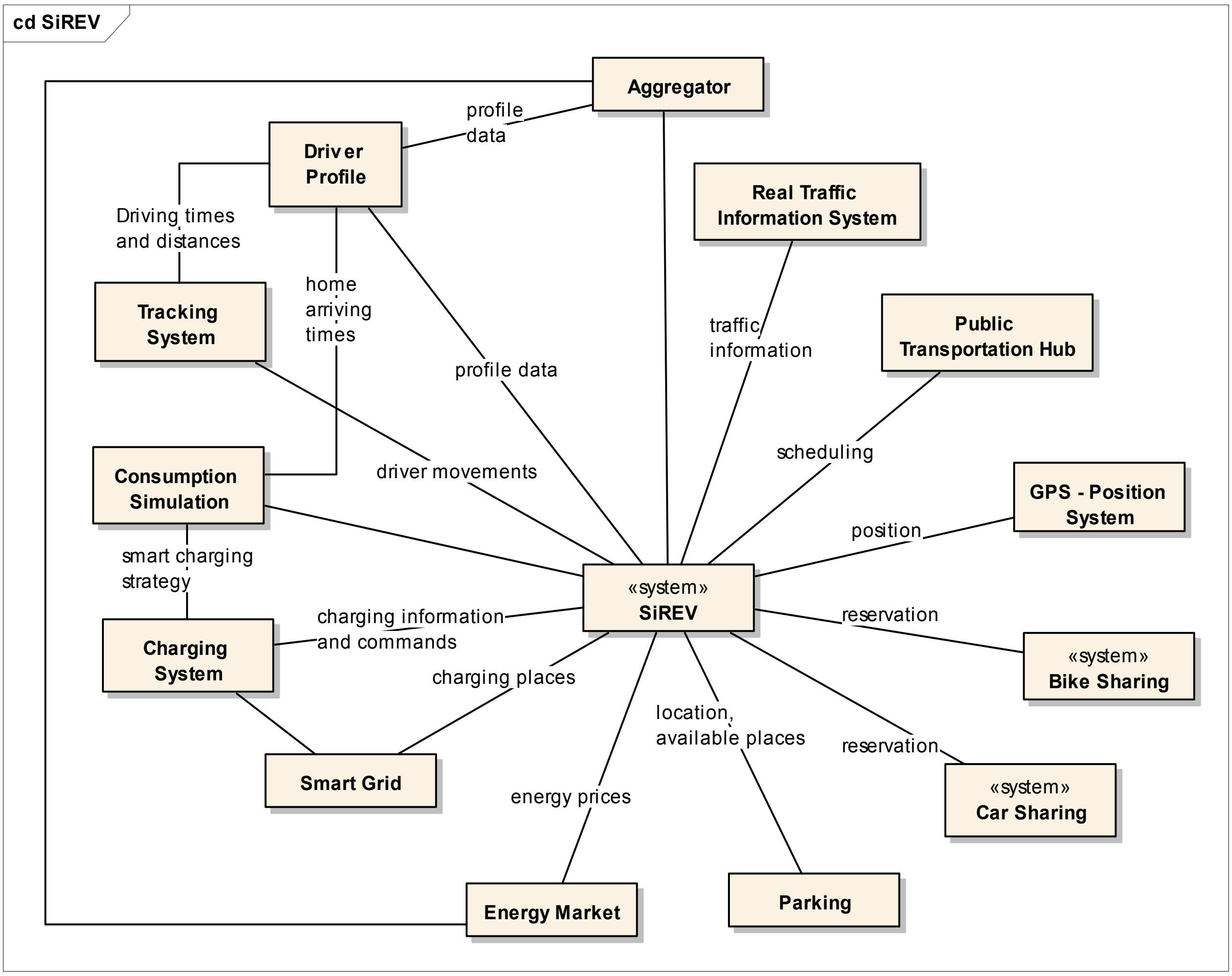

Fig. 1. Main modules of Mobi_System.

- Mobility Sustainability functions: (1) Real Time Traffic Information System - a system to integrate several operator traffic sensors information that are available. (2) Public Transportation Hub - a system with public transportation routes and schedules. The receiving of such information will be made by consulting a file in XML format that contains information prepared and compiled by different operators. This file includes the geographical location of sites where the user can embark on public transport. The public transport information will be incorporated into items of candidate recommendation system, by allocating an item property that indicates whether or not it is close to public transport. This information is another dimension that enters on the calculation of the usefulness of each item which positively or negatively affects the score according to the choices that were presented to the driver. (3) Parking Places Available. (4) Information on Points of Interest - Information about points of interest are preloaded on the system and is used for direct consultation by the driver who can perform a quick search for points of interest near the present location. The information is also used in the recommendation of charging points in the Mobi_System. In this situation, before the candidate items are sent to the recommendation system, those are processed against the information of points of interest, and all that remain at a distance of less than 5 km from a point of interest will be marked as being "near the point of interest". (5) Car Sharing and Bike Sharing - systems to support mobility functions in 'smart' cities. These functions are described in [1].
- Energy Market Functions (described in detail in [2, 3]): (1) Aggregator for energy market participation. (2) Energy Market Prices. The SiREV (System of Recommender for Electric Vehicles) consults the energy market regularly, and obtains the discounted value of the price of electrical energy. The only information received is the fare in the current format €/kWh, or euro value per thousand watt hours consumed. The energy market information is used to control the batteries system of the electric vehicle during loading periods. If the price of energy rises above a configurable threshold, the SiREV sends a command to the batteries system to stop charging. If the price falls back, it is sent a command to start charging again. (3) Consumption Simulation – developed for smart charging strategies taking into account electricity distribution network limitations (for more detail, please see [2]).

III. MOBI_SYSTEM ARCHITECTURE

From Figure 2 it can be seen the different modules of the proposed application: (1) Application for Mobile Devices; (2) Web Service EVService; (3) Services Layer (Service Layer); (4) Data Access Layer (DAL); (5) SGBD1; (6) SGBD2 and the Recommendation System (Recommender Engine), responsible to find user relevant information. To isolate the calculation of the recommendations and give some scalability to the system, it were used two DBMSs (database management systems): SGBD1 and SGBD2. The SGBD1 owns all of the data stored by the Mobi_System, while SGBD2 has data that are needed for the calculation of relevant information. These two DBMSs perform data replication, so that the calculations of the recommendations on data are consistent and fair. The (Data Access Layer) performs the data management of the Mobi_System, and it operates on the SGBD1 questions in the form of SQL (Structured Query Language). This provides a centralized access and ensures safety and consistency of the database SGBD1. Interaction with the data access layer is done using DTOs (Data Transfer Objects). DTOs are a design pattern used to transfer data between subsystems of a software application. The (Service Layer) is the business layer of the Mobi_System, and it is where the services are provided by the centralized system. Among the many services there are services related to the loading points (where the batteries of the electric vehicles are charged), as the researching and booking of loading points. To expose the system services of the Mobi_System, it is used the Web Service EVService. Web Services are a solution used in systems integration and communication between different applications, enabling new applications to interact with existing ones. Mobile App is an application for mobile devices whose operating system is based on the set of Windows CE components, such as Windows Mobile 6.0. This application accesses the system services through the Web Service EVService using http requests (Hypertext Transfer Protocol).

The data base performed in SQL have the follow entities: (1) User - contains the minimum information of the user; username, password, name; (2) Profile - contains data for the profile: birth date, gender, nationality; (3) Category - represents a category; (4) Paper (Role) - set of permissions that are associated to the users; (5) Classification - represents a classification with a name and value; (5) Points of Interest (POI) - contains information specific to a place of interest; (6) Loading Points (CP) - contains information specific to loading points; (7) Slot - time interval used to schedule reservations; (8) Period - time interval, used to store the operating time for local; (9) Schedule - time interval used to divide periods; (10) Relevant Information (Prediction) - stores the last generated recommendations; (11) Local - represents a place, represented by its location (latitude and longitude, in decimal), for his address, name, image presentation and a brief description; and (12) Public Transportation Information.

IV. REAL TIME INFORMATION AND BEST PATH

To determine the best path between two points on a map, it creates a graph representing the map, where the arcs represent roads and nodes represent intersections or traffic areas. It applies an algorithm on the graph to find the path with less weight and faster between the two desired points.

The arc weight is determined based on the distance of start and end node and the maximum speed allowed. Real time traffic information adjusts this speed. So the weight of an arc is basically the average time in seconds that is needed to be traveled. The expression used in the calculation is:

$$arc\ weight = 3600 \,.\, L \,/\, V \tag{1}$$

Where L represents the length of the graph, in km (distance from start to end node), and V is the average speed, in km/h, taken from real traffic information system. For all arcs there is a speed limit which serves as the basis for the weight of this case when there is no traffic information. When there is traffic information, for a particular arc, the weight of this is affected because the traffic influences the speed. For example, if an arc with heavy traffic has a speed limit of 90 km/h, given the existing traffic, the reduction factor of the maximum speed will have a value set between 0 and 1, based on current traffic information, causing the increase of the time required for this arc.

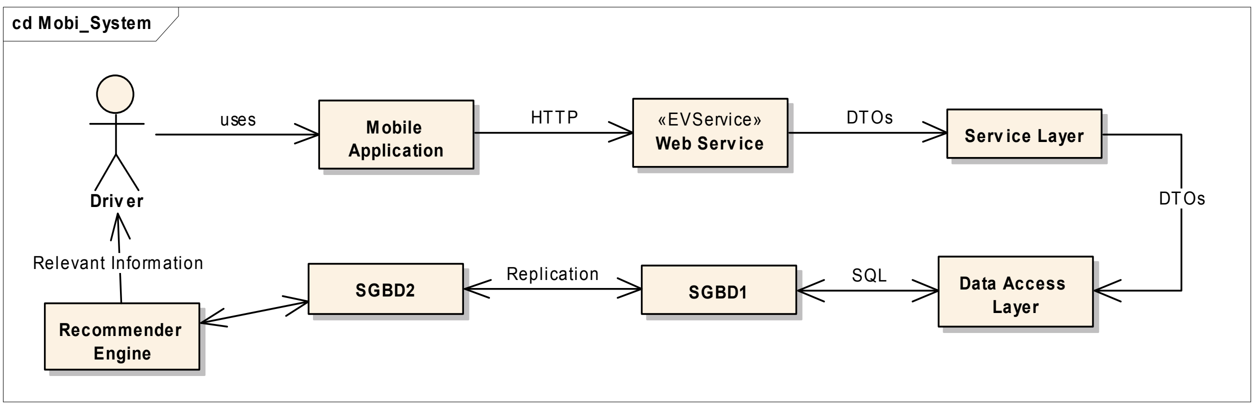

Fig. 2. Architecture of Mobi_System.

If the transit is cut off, the factor of speed reduction will have the value 0, and the maximum attainable speed for low 0 km/h that causes this arc has infinite weight. If traffic is proceeding smoothly and without any problems, the factor of speed reduction have a certain value, and a maximum attainable speed becomes equal to the speed limit, which makes this arc have the lowest weight possible, providing a reduction in the mean time the arc needs to be traversed. This weight can integrate also public transportation information and a price associated with CO_2 emission of private transportation. Users can choose the impact parameters based on their strategy: saving money, saving time, or even convenience and comfort. Please, see [5] for a complete description of this process.

Example: Path A1 (city of Alhandra – city of Alverca) - Distance is 5 km (*L*), maximum speed allowed on this section is 120 km/h (in motorway), but traffic information shows average speed is 84 km/h (*V*), so, according to (1):

arc weight = 3600 . (*arc distance* / *average speed*) = 214 s.

Let's consider that public transportation takes more or less two times more to travel the same path: 400 s. Transportation price is 1 € and car transportation is 2 €, including the price for CO_2 emissions and motorway toll price. If drivers choose the same weight factor for time and price, then the new arc weight for the car is 214 x 2 = 428 s, while the arc weight for the public transportation is 400 x 1 = 400 s. In this case the arc weights are approximately the same, but if we also put the price for parking the car in the city, it will increase again the arch weight for private car, and then the system will suggest the public transportation as a better option. In this case the system shows transportation time table and gives orientation to reach transportation stop using Google Maps interface.

To determine the best route or the quickest route was implemented the Dijkstra's algorithm, where its running time is proportional to N2, with N being the number of nodes in the graph. The graph to be used by the Dijkstra's algorithm is represented by an xml file. In Mobi_system was created a class called Dijkstra, which is where the algorithm is implemented. In the implementation were used the following data structures: (1) An array of nodes (nodes) that have all the nodes in the graph, see Figure 3; (2) A list of integers (unsettledNodes), which holds the identification (id) of the nodes for which it has yet found the best path to the source node; (3) A list of integers (settledNodes), which the id of the nodes already found the best path to the source node; (4) A Dictionary (shortestDistances), which is the key to an integer representing the identifications (id's) of nodes, and that has a value (another integer that represents the best way to estimate the source node); (5) A Dictionary (predecessors), which is the key to an integer representing the id's of nodes, and which is another integer value that represents the id's of nodes in the previous best path to the source node. For a complete description method, please see [5]. Information exchanges with external systems are based on XML files.

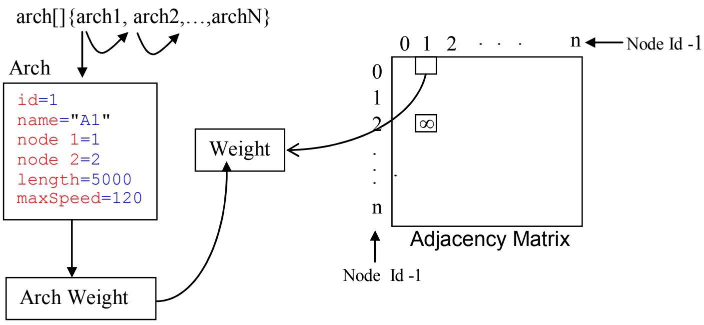

Fig. 3. Filling Adjancy Matrix.

V. IMPLEMENTATION

For a complete description of the implementation, please, see [3]. The (Service Layer) is a layer that is intended to expose business logic as services and thus serve as an abstraction of business logic. This layer was created and designed so that a future Web application can take advantage of the functionality of the system. These are services that perform operations on the system being Mobi_System that the Web Service EVService uses this layer to perform the operations on the Mobi_system. These services do not show any relation between them, since their functioning are independent of others and consist of: (1) Service Loading Points (IRechargingPointsService); (2) Service Sights (ISightsService); (3) Service Recommendations (IRecommendationsService); (4) Service Ratings (IClassificationsService); (5) Service Directorates (IDriverDirectionsService); (6) Map Service (IMapsService); (7) Service Users (IUsersService); (8) Service Roles (IRolesService); (9) Service Categories (ICategoriesService); (10) Geocoding Service (IGeocodingService); (11) Car and Bike sharing service; and (12) Public transportation service.

Creating Functionalities Based on Existing Services

With this solution developed it is very simple to add services and functionalities to the system, since the solution was developed in a modular and incremental form.

For instance, given the suggestion of a set of users that it would be interesting to have information on the location and availability of parking places, this functionality is fairly easy to add, which allowing users to obtain directions to car parks in town. It could also be possible, in a effortless way, to get information on the availability of car seats (it would be an operator responsibility to manage the parking places and car seats availability, using some format data). Similarly, it is possible to interact with the application, managing the car x to the park y, and it is possible to reserve car seats and get guidance until the car park. For this task you need it is necessary to: (1) Add to the local authority car parks to the conceptual model of the database; (2) Utilize or even replicate (in large part) the service of loading points; (3) Provide Web Service operations in EVService; and (4) Add a set of screens for the car parks in the application for mobile devices.

The same principles principals here described were used to develop car and bike share systems, which are described in [1].

VI. Case Study

The driver of the EV can use the proposed Mobi_System to: (1) Get directions; (2) Locate loading points (for charging the batteries) and reserve parking lots slots, (3) Get recommendations on his journey with varied important information, like: batteries charging, whose handicaps are known involving the loading time and their autonomy, local information satisfaction, public transport information in case of car failure, etc; (4) Points of interest; (5) Smart charging strategy; and (6) Electrical market participation (best periods to sell or buy electrical energy, in order to obtain profit maximization).

After user login (authentication), there is a menu where he selects which operation he wants to make, as shown in the left side of Figure 3. If he wants to know possible interest places, from Recommendations tab Places it is opened the screen shown in the right side of Figure 4. The user defines which region the system should consider, introducing the geographic center and the radius, which can be obtained using GPS or an address. Pressing the button Get, recommendations are obtained, in this case the user wants to know places of interest near the city of Seixal, in Portugal, considering a radius of 2 km. At the bottom of the screen you he will find the recommendations identified by the name and address of the location of interest. In the Option Menu, which is in the lower right corner, it contains the following: (1) Info; (2) Classify; and (3) Map. The user can get more information about the location of interest selected by pressing the Info option from the menu. In the following figures it can observed how is the information is shown for a point of interest (POI).

Information from a point of interest is constituted by its name, address, latitude, longitude, brief description, associated categories, hours of operation and a display picture. In the user classification menu option, the user can rate the place of interest in order to help future recommendations to him and to other users. The screen allows the user to annotate/classify the point of interest with a degree of satisfaction, as presented in the left side of Figure 4 (where it shows the rate “Bom” in Portuguese language, meaning “Good”). If the user wants help geographically he can access it from the Map menu to display the selected point of interest, as is visible in the center of Figure 4. The site contains an identifier of interest (in this case is A), and in this screen the user has the ability to navigate the map, as well as zoom in and zoom out. In this menu the user can change the map type view (Hybrid, Road Map, Satellite). If he doesn’t like the recommendations generated, he can always get all the sites in a given area by accessing the item in the Places tab View. This screen is similar to the screen that lets the user gets recommendation of sights, but this shows all the sites in a given region with the possibility of filtering according to the categories of points of interest, as shown in the left side of Figure 5. After the user selects possible points of interest to visit during the holidays, he needs to know which route to use in order to visit these places of interest. For that he can use the screen which is on the Path tab in the Search option. This screen is shown in the right side of Figure 5. On this screen the user enters the start and the end places and current location is taken from GPS. After pressing the button Get Directions, the application indicates possible routes and also indicates the travel time and distance. For each route it is possible to see the steps needed to guide the user during the journey in order to reach the target site. Optionally, the user can observe this route on a geographical map.

After the user gets the recommendations and the routes he has to follow, he needs to know where he can recharge the batteries (loading points) during or after the path followed. To access this screen the user has to choose the option Book tab Recharging Stations. The screens related to the availability and reservations of loading points (recharging stations) are visible in Figure 6. On the screen shown in the left side of Figure 7 the user identifies in which area he wants to find the loading points, and then he presses Search to locate the application available loading points. After that, the user selects the loading point that he wants to use, and may view the information of the Info button, the geographic location in the Map button, and can get an slot available at the recharging station using the Get button. The user can restrict the range of available slots by setting the start time and the end time, visible in Figure 8. The Book button may submit a request to reserve a given slot, after which the user you should receive a notification with the response of the order (not implemented). Example of Submissions and Reservations are shown in Figure 6.

A further usage possibility for these functionalities is available in [4].

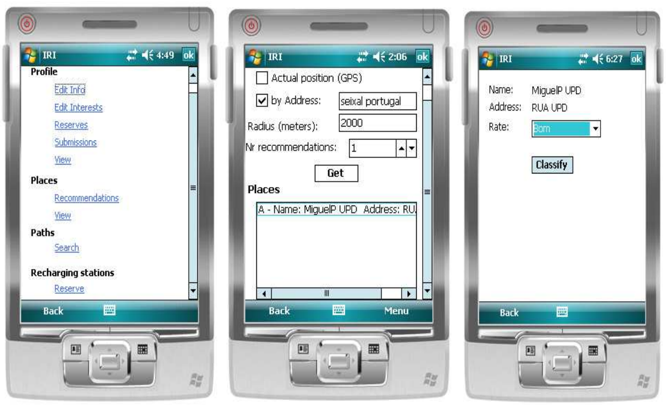

Fig. 4. Main application screen, search and classification of POI.

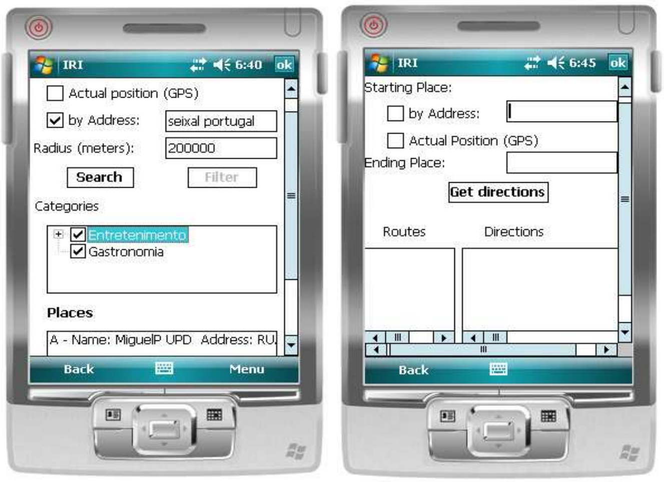

Fig. 5. Get POI and *Get* directions.

Fig. 6. Availability and Reservation of slots at recharging stations.

VII. CONCLUSIONS

This paper describes the work that has been developed in order to provide an application (for mobile devices) that can be used by electric vehicles drivers during their journeys. The application provides varied and important information, like the display of the closer loading points (for recharging the car batteries), recommendation or consultation of points of interest, car and bike sharing reservation, and public transportation information.

The design of this application presented many difficulties, specifically in terms of design. The main development idea was a flexible and reusable application, because this is a starting idea and it is foreseen that several changes will be performed. Another feature of the mobile application is the use of GPS, which can be critical as an application for the driver, but that has added value to the work and to the developed efforts.

One of the features of the system is the recommendation of points of interest. The recommendation process aims to provide the driver with information on near places of interest that the application considers appropriate for the driver, based on his interests and on his voting (ratings).

The solution developed is still a prototype, since electric vehicles are at an early stage of development, and thus, the implementations of many changes to this application are already expected.

ACKNOWLEDGEMENT

The authors are grateful to FCT (*Fundação para a Ciência e a Tecnologia*) for funding the Project PTDC/EEA-EEL/104569/2008 and to FTC and MIT-Portugal Program, for funding Project MIT-PT/EDAM-SMS/0030/2008.

REFERENCES

[1] João Ferreira, Paulo Trigo and Porfírio Filipe. “Collaborative Car Pooling System” Proceedings of International Conference on Sustainable Urban Transport and Environment, Paris, 24-26 June 2009.

[2] Vítor Monteiro, João C. Ferreira, Gabriel Pinto, Delfim Pedrosa, João L. Afonso, “iV2G Charging Platform”, IEEE-ITSC, 13th International IEEE Conference on Intelligent Transportation Systems, 19-22 September 2010, Madeira, Portugal. IEEE Catalog Number: CFP10ITS-CDR, ISBN: 978-1-4244-7658-9, ISSN: 2153-0009, pp. 409-414.

[3] João C. Ferreira, João L. Afonso, “A Conceptual V2G Aggregation Platform”, EVS-25 -The 25th World Battery, Hybrid and Fuel Cell Electric Vehicle Symposium & Exhibition, 5-9 Nov. 2010, Shenzhen, China.

[4] Luís Borges and Miguel Borges. IRecommendIt: Sistema de recomendação para veículos eléctricos. End year project at ISEL.

[5] Tiago Fernandes and Miguel Fontes. Determinação do melhor caminho em sistemas de transporte terrestre (BPath – Best Path). End year project at ISEL.